

1

2 **Supplementary Information for**

3 **Understanding the Role of Individual Units** 4 **In a Deep Neural Network**

5 **David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, Antonio Torralba**

6 **Corresponding author: David Bau.**

7 **E-mail: davidbau@csail.mit.edu**

8 **This PDF file includes:**

- 9 Supplementary text
- 10 Figs. S1 to S18
- 11 Captions for Movies S1 to S3
- 12 References for SI reference citations

13 **Other supplementary materials for this manuscript include the following:**

- 14 Movies S1 to S3

15 Supporting Information Text

16 **A. Emergent concepts across layers and networks.** We compare emergent concepts that appear across more layers and
17 architecture variations for scene classification CNNs, and we compare emergent object concepts across layers in the GAN
18 generators trained with different scene training set.

19 Figure S1 compares the emergence of units that match semantic concepts in classifiers trained with different architectures
20 (VGG-16 (1), AlexNet (2), and ResNet (3)) and in generators trained on different data sets (LSUN kitchens, bedrooms, and
21 outdoor church scenes (4)). A similar pattern of detected concepts can be seen across these variations: in the classifiers, the
22 largest number of detected objects appears in the final convolutional layer, while in the generators, units that match most
23 objects appear in the middle layers of the network.

24 Figure S2 details the concepts that are matched at every convolutional layer of VGG-16 and AlexNet.

25 Figure S3 characterizes texture-sensitivity and shape-sensitivity of units by observing changes in behavior each unit on
26 a stylized images in which textures have been randomized. We construct a stylized-places data set using the stylization
27 method described in (5). Then $IoU(s, u)$ denotes the IoU when comparing the 99% activation of unit u on the original
28 Places images compared the behavior of the same unit on the stylized-Places images (indicated as s). Low $IoU(s, u)$ indicates
29 texture-sensitivity and high $IoU(s, u)$ indicates shape-sensitivity. Figure S3a-c show the distribution of texture-sensitive versus
30 shape-sensitive units across layers of VGG-16. Figure S3d visualizes texture-sensitive units which activate on very different
31 parts of images when the style is altered, and Figure S3e visualizes shape-sensitive units which activate in many of the same
32 parts of images when the style is altered.

33 **B. Characteristics of the most- and least- important units.** We further explore the role of class-important units within a classifier,
34 asking how much single-class accuracy can be improved by dropping unimportant units; go what degree units that are important
35 to multiple scene classes match lower-level semantic visual concepts closely; and whether important units are positive correlated
36 or negatively correlated with their associated classes.

37 Figure S4a-b show the increase in single-class classification accuracy, by eliminating the 256 `conv5_3` units that are least
38 important to the class. Eliminating these units improves the network’s ability to classify specific class, increasing accuracy
39 from 76.6% on average to 92.1% on average. This effect has been observed across all classes.

40 Figure S4c-f explore the relationship between the number of scene classes for which a unit is important and the mean IoU of
41 the unit. Units that are important to many scene classes have higher IoUs (i.e., they more closely match a semantic visual
42 concept such as an object or an object part) when compared to units that are important to none or few scenes. In these
43 plots, the relationship is shown to hold, regardless if a unit is counted as important to a class by when it is among the top-2
44 important, top-3, top-4, or top-5.

45 Figure S4g, reveals a relationship between the mean class importance of a unit and its IoU. The mean class importance of a
46 unit is the reduction in single-class (binary classification) accuracy when the single unit is removed from the network, averaged
47 over all the output classes. Note that mean class importance is different from the reduction in multiclass all-class classification
48 accuracy when a unit is removed: Units that are important to several classes will tend to have higher mean class importance,
49 but units that contribute to multiclass classification accuracy without being important to any class will tend to have lower
50 mean class importance. Figure S4g divides the 512 units into 16 buckets of 32 units, grouped by mean class importance, and
51 shows that the most important units tend to have the highest mean IoU, i.e., they most closely match semantic visual concepts.

52 Figure S4h examines the correlation between each unit and each class, comparing units that are important to each class with
53 other units. The plot includes 512×365 points, one for the relationship between each of 512 units and 365 output classes. The
54 importance is plotted as the reduction in single-class accuracy when the unit is removed from the network, and the correlation
55 relates the peak activation of the unit to the softmax prediction score of the class. The top-4 most-important units for each
56 class are colored blue; the remaining relationships are orange. The highest density of the plot is enclosed in the red box: 58.9%
57 of unit-class correlations are negative. However, almost all correlations for top-4 units for each class is a positive correlation.
58 This would not necessarily need to be the case: for example, the network could in principal encode a rule such as "grass never
59 appears inside a conference room" by giving importance to a grass-detecting unit that negatively correlates to the conference
60 room class. However, such negative signals do not seem to be important: all the important relationships between `conv5_3`
61 units and output classes have positive correlations.

62 Figure S5 shows several examples in which different overlapping subsets of units are important to different output classes.
63 In each of the examples shown, there are three units that are each within the top-4 most important units for two out of three
64 of the three output classes, but not the third. For example: a ‘house’ detector unit is important to both the ‘boathouse’ and
65 ‘farm’ class, but a ‘water’ unit is only important to ‘boathouse’ and a ‘grass’ unit is only important to ‘farm’; in contrast,
66 ‘water’ and ‘grass’ units are both important to the ‘pond’ scene class, whereas the ‘house’ unit is not important to ‘pond’.
67 These varying combinations illustrate the effects seen numerically in Figure S4c-f: units that are important to multiple different
68 output classes tend to align with meaningful concepts that can be used to make distinctions that generalize beyond one class.

69 **C. Full visualization of all the units in a layer.** We expand upon the selected visualizations of units shown in the main paper by
70 providing the same visualization for all the units in the analyzed classifier and generator layers.

71 Figure S6 through Figure S12 show visualizations of all 512 units of `conv5_3` of the VGG-16 tested in the main paper. Each
72 unit is illustrated with the top-5 activating images from the validation set, highlighting the regions where the activations exceed
73 the 1% quantile. For each unit, the best-matching concept with the highest $IoU_{u,c}$ is shown, along with all output scene classes

74 for which the unit is among the top four important units for the class. Units are listed in order from the highest *IoU* to the
75 lowest.

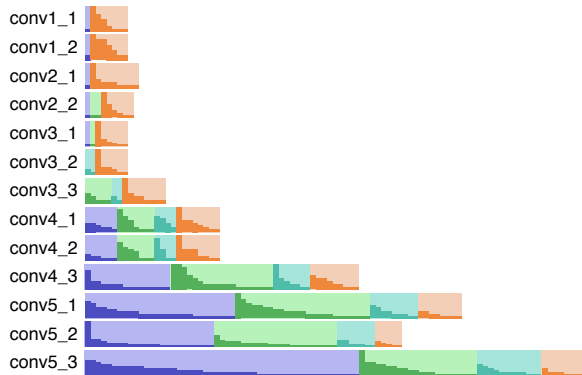
76 Figure S13 through Figure S18 show visualizations of all 512 units of `layer5` of the Progressive GAN kitchen image
77 generator (6) tested in the main paper. Each unit is illustrated with the top-5 activating images from a set of 10,000 generated
78 images, highlighting regions where the activations exceed the 1% quantile. For each unit, the best-matching concept with the
79 highest $IoU_{u,c}$ is shown.

80 **D. Video demonstration of GAN image manipulation.** Our provided image-editing movies show direct manipulation of units of
81 a generator. In Movie S1, trees are removed from scenes, revealing details behind the trees. In Movie S2, doors are added to
82 scenes, showing context-sensitivity. In Movie S3, a variety of other edits are done, showing creative effects that can be achieved
83 by allowing a user to manipulate units directly.

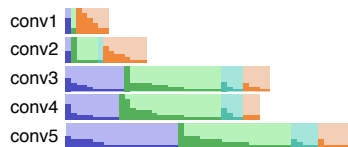
84 References

- 85 1. K Simonyan, A Zisserman, Very deep convolutional networks for large-scale image recognition in *ICLR*. (2015).
- 86 2. A Krizhevsky, I Sutskever, GE Hinton, Imagenet classification with deep convolutional neural networks in *NeurIPS*. pp.
87 1097–1105 (2012).
- 88 3. K He, X Zhang, S Ren, J Sun, Deep residual learning for image recognition in *CVPR*. (2016).
- 89 4. F Yu, et al., Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint*
90 *arXiv:1506.03365* (2015).
- 91 5. R Geirhos, et al., Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness.
92 *arXiv preprint arXiv:1811.12231* (2018).
- 93 6. T Karras, T Aila, S Laine, J Lehtinen, Progressive growing of gans for improved quality, stability, and variation in *ICLR*.
94 (2018).

(a) vgg16 classifier, from Fig.1



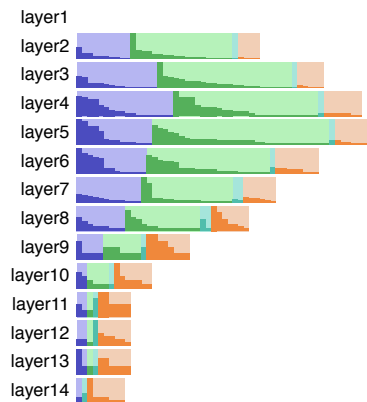
(b) alexnet classifier



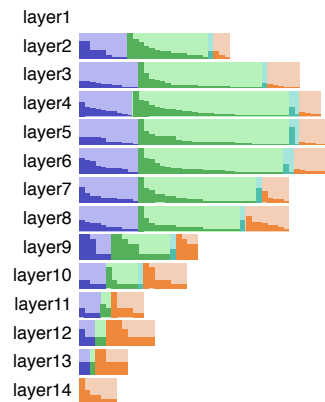
(c) resnet152 classifier



(d) kitchen generator, from Fig.3



(e) living room generator



(f) outdoor church generator

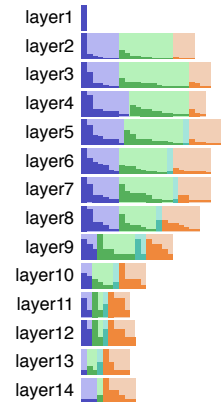


Fig. S1. Comparing layers across models. (a) For reference, the VGG16 classifier (1) from the main paper Fig.1. (b) An AlexNet classifier (2) trained on the same scene classification problem. A similar structure emerges, but fewer object classes. (c) A ResNet152 classifier (3) trained on the same problem. The structure is similar again. Each layer of ResNet is a residual block consisting of many convolutions: in ResNet152, layer4 contains 105 convolutions that compute residuals that are added together. Only non-residual layer representations are shown. (f) For reference, the progressive GAN model trained to generate kitchen images, from the main paper Fig.3. (g) A progressive GAN (6) trained to generate living room images shows a similar structure. (h) A progressive GAN trained to generate outdoor church images represents a smaller number of identified classes, but the same overall structure, with object-specific units peaking at the middle layers.

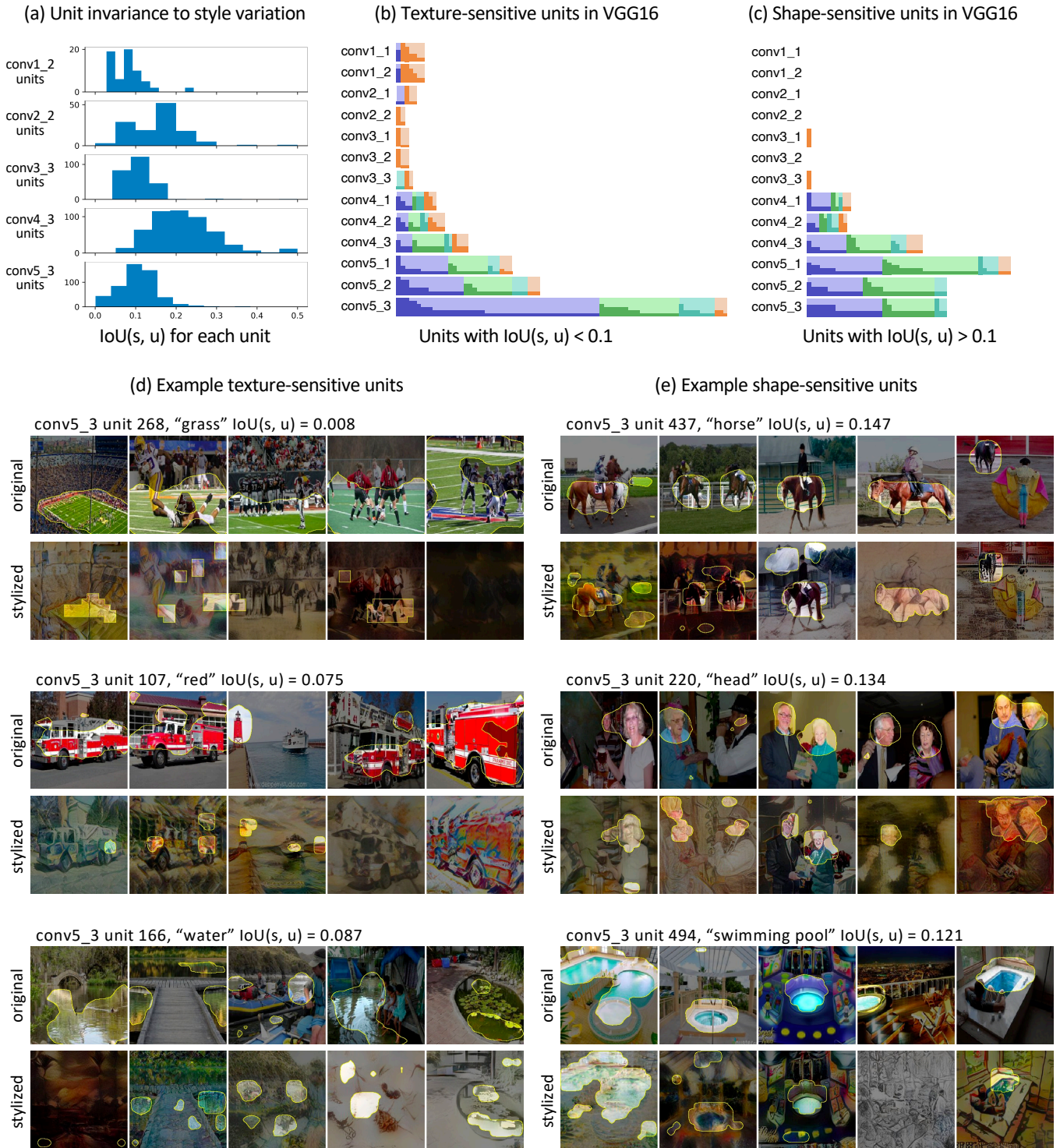


Fig. S3. Characterizing texture-sensitivity and shape-sensitivity of units by observing changes in behavior each unit on a stylized images in which textures have been randomized. (a) The distribution of $IoU(s, u)$ on different layers of VGG-16 trained on places. All layers show many texture-sensitive units including the final feature layer, but some shape-sensitive units emerge at later layers such as conv4_3. (b) Texture-sensitive units for every layer, plotted as in Figure S2; only units with $IoU(s, u) < 0.1$ are shown. (c) Shape-sensitive units plotted, limiting to units with $IoU(s, u) > 0.1$. (d) Visualization of texture-sensitive units: when the style is changed, they activate on different portions of the images. (e) Visualization of shape-sensitive units: even when the style is changed, these activate on many similar regions within images.

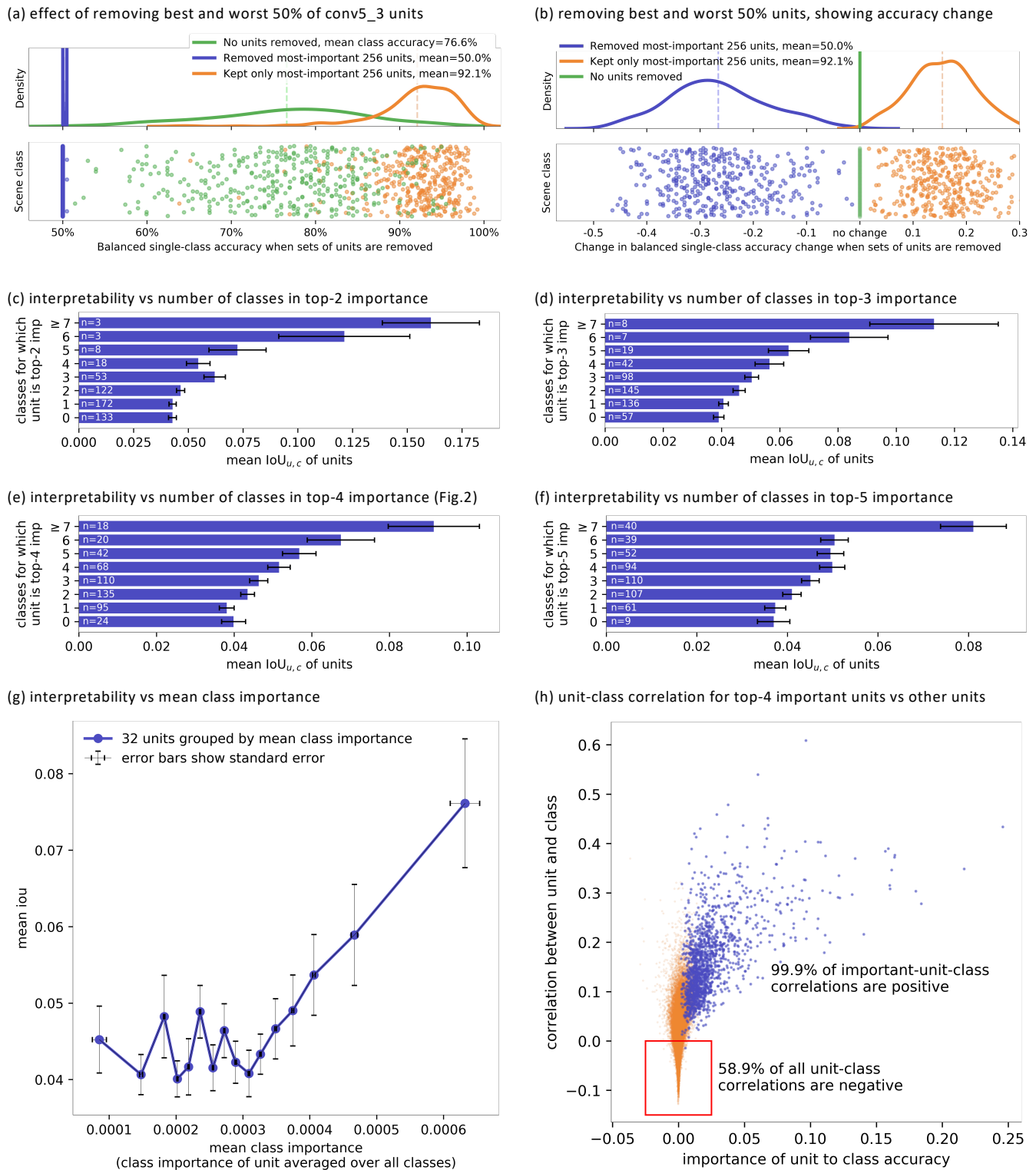


Fig. S4. (a) The effect of removing the least-important 256 (out of 512) units for each class. Although no retraining is done, removal of these units increases single-class binary classification accuracy for the chosen class from an average of 76.6% to 92.1%. (b) The same data plotted as changes from the original network accuracy on the class, showing that removing the most-import 256 units always damages accuracy and removing the least-import 256 units always improves accuracy for the class. (c-f) Mean $IoU_{u,c}$ is higher for units that are important for more output scene classes; this finding is insensitive to the way we count importance. (c) shows the results for top-2 importance of units; (d) top-3; (e) top-4; (f) top-5. (g) The mean class importance of a unit is the single-class accuracy drop caused by the removal of the unit averaged over all 365 classes. When units are bucketed by mean class importance, the most-important units more closely match human-interpretable concepts as measured by higher IoU. (h) Importance and Pearson's correlation between activation and class for top-4 important units for each class, plotted in blue; less important units for each class are in orange. While most unit-class pairs show a negative correlation, almost all unit-class correlations for important units are positive. The network learns no specific negative correlations.

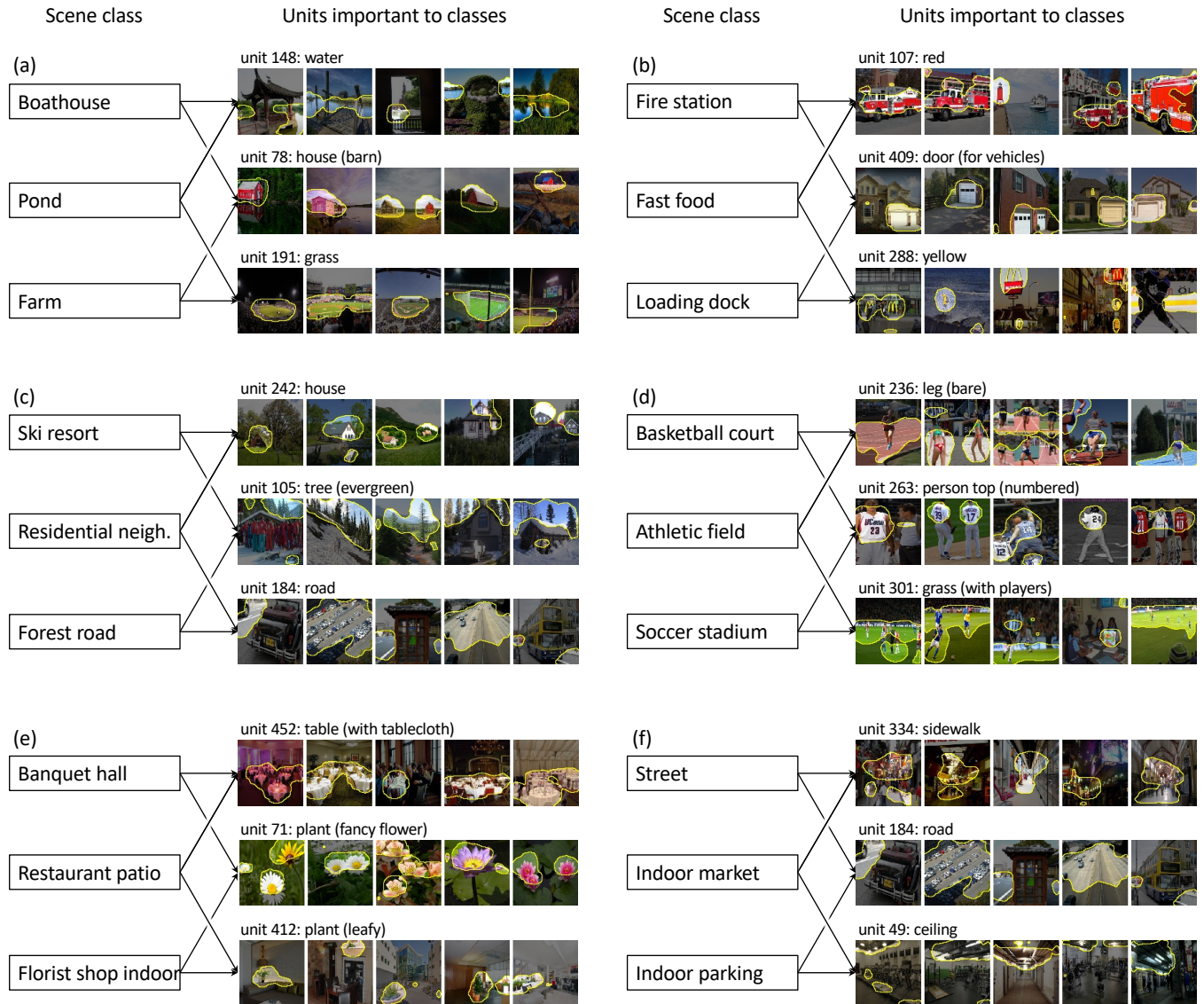


Fig. S5. Combinations of important units that contribute to an output class reveal the structure of the classification model learned by the network. Here we show that a different subset of three units is important to different output class. (a) A boathouse is a barn on the water; a pond is water with a grassy field; a farm is a grassy field with a barn. (b) yellow and red without a garage door signal fast food; red with a garage door is a fire station; yellow with one is a loading dock. (c) two important ski resort units are also important for residential neighborhoods and forest roads, but those classes also need to see roads. (d) three sport fields can be distinguished: bare legs are seen in basketball and athletic fields but not soccer; numbered jerseys show up in basketball and soccer but not athletic fields; and basketball courts do not have grass. (e) the distinction between a banquet hall and a restaurant patio is the type of plants, which can be seen in visualization although the segmentation does not distinguish them: flowers or leafy plants. (f) the presence of a ceiling indicates an indoor parking lot or an indoor market rather than a street scene.

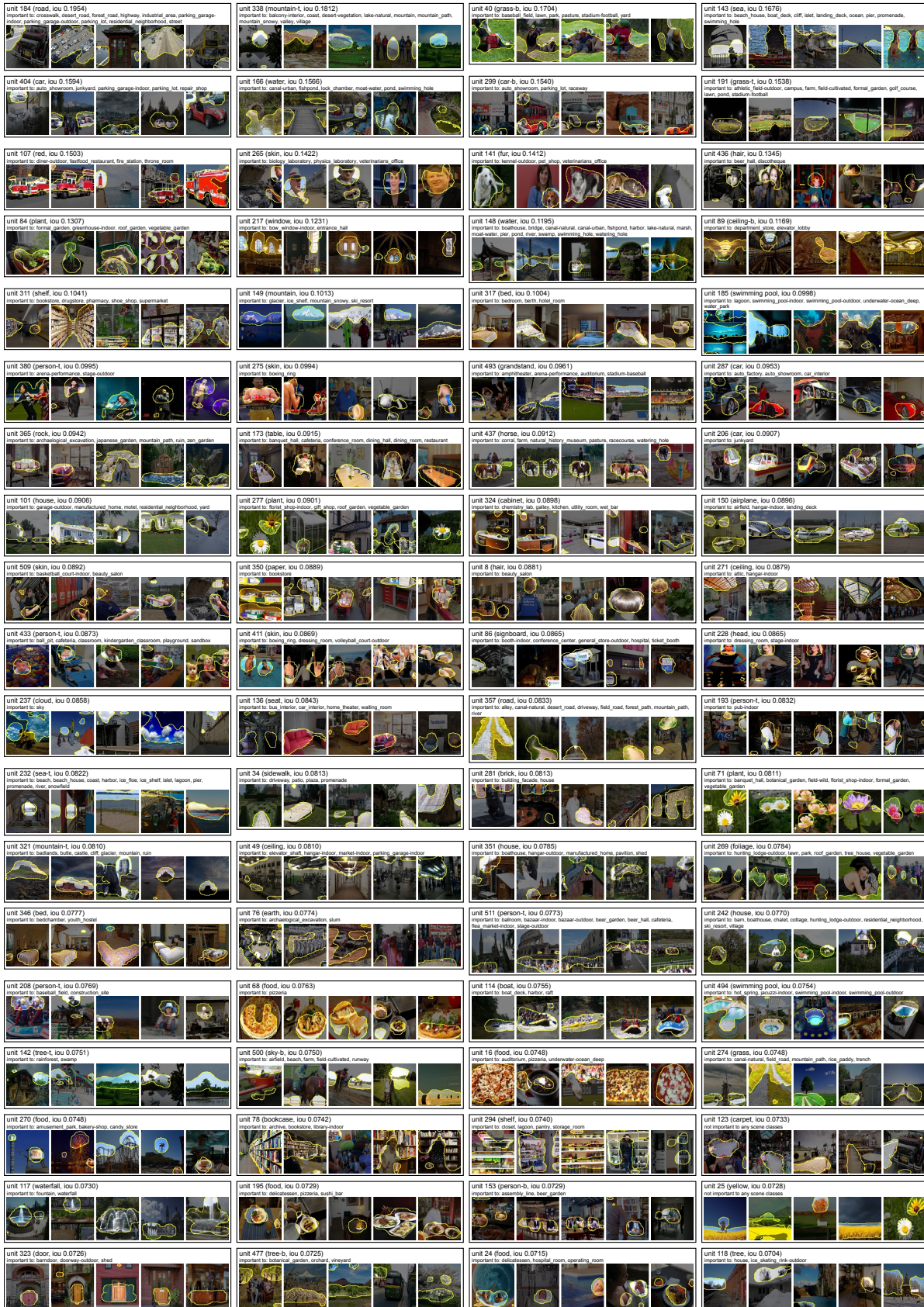


Fig. S6. Comparing all units of conv5_3 of VGG16, 1 of 7. For each unit, the top five activating images are shown from the Places365 validation set, highlighting regions where the activation spikes above the top 1% quantile. Each unit is listed with the best-matching segmented concept and all output scene classes for which the unit is among the top four important units. Units are sorted in descending order of IoU agreement with a segmented concept.

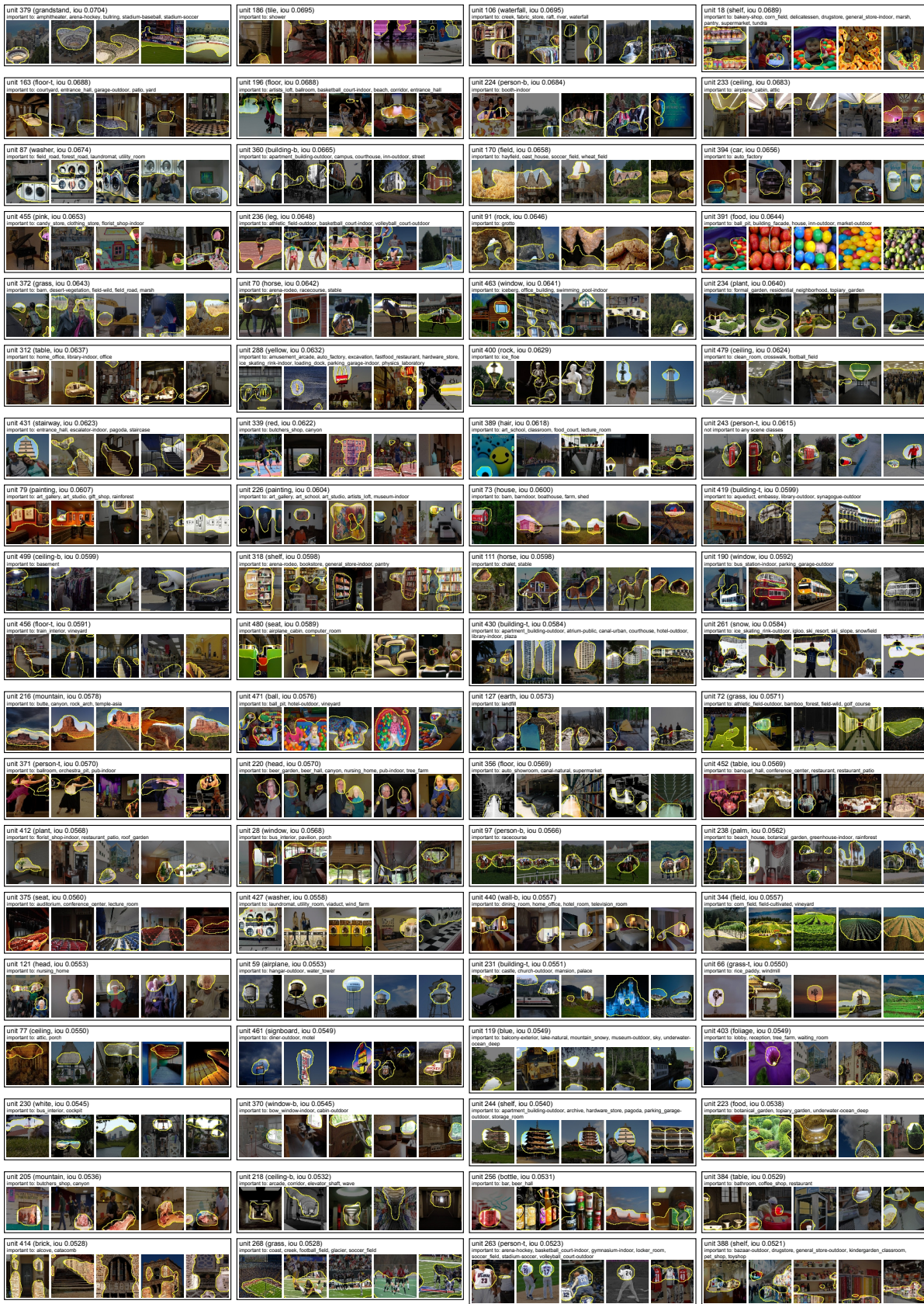


Fig. S7. Comparing all units of conv5_3 of VGG16, 2 of 7.

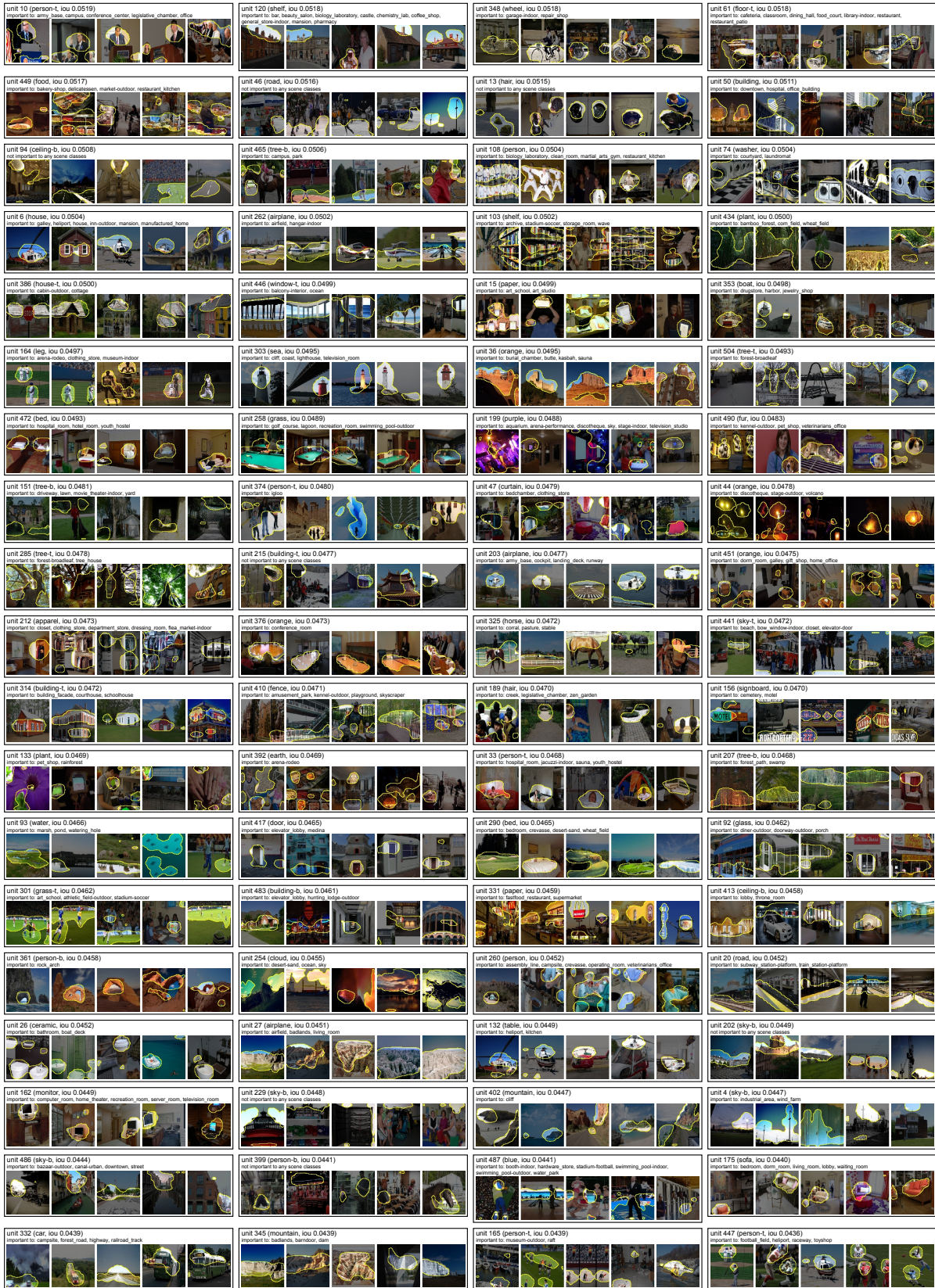
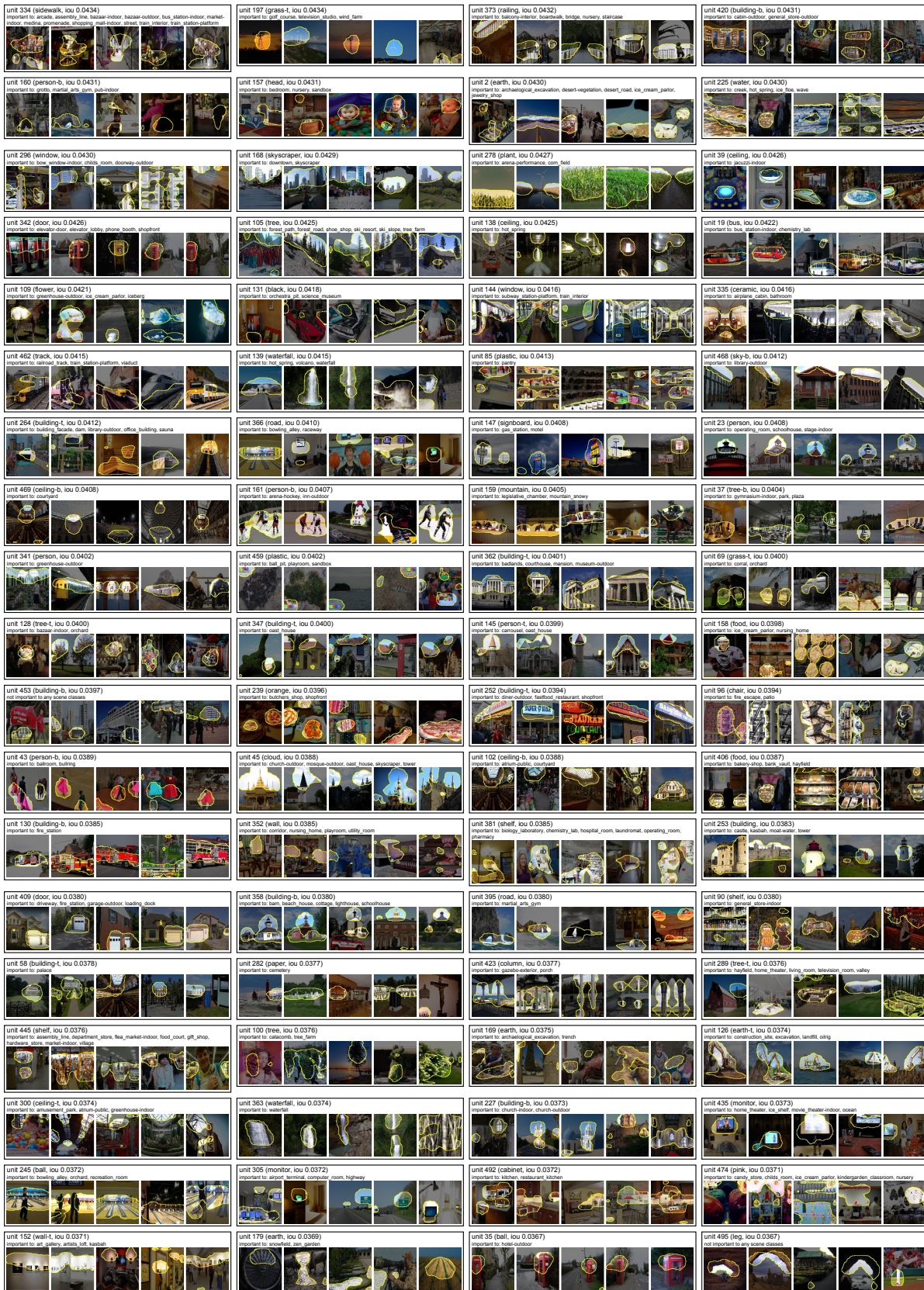


Fig. S8. Comparing all units of conv5_3 of VGG16, 3 of 7.



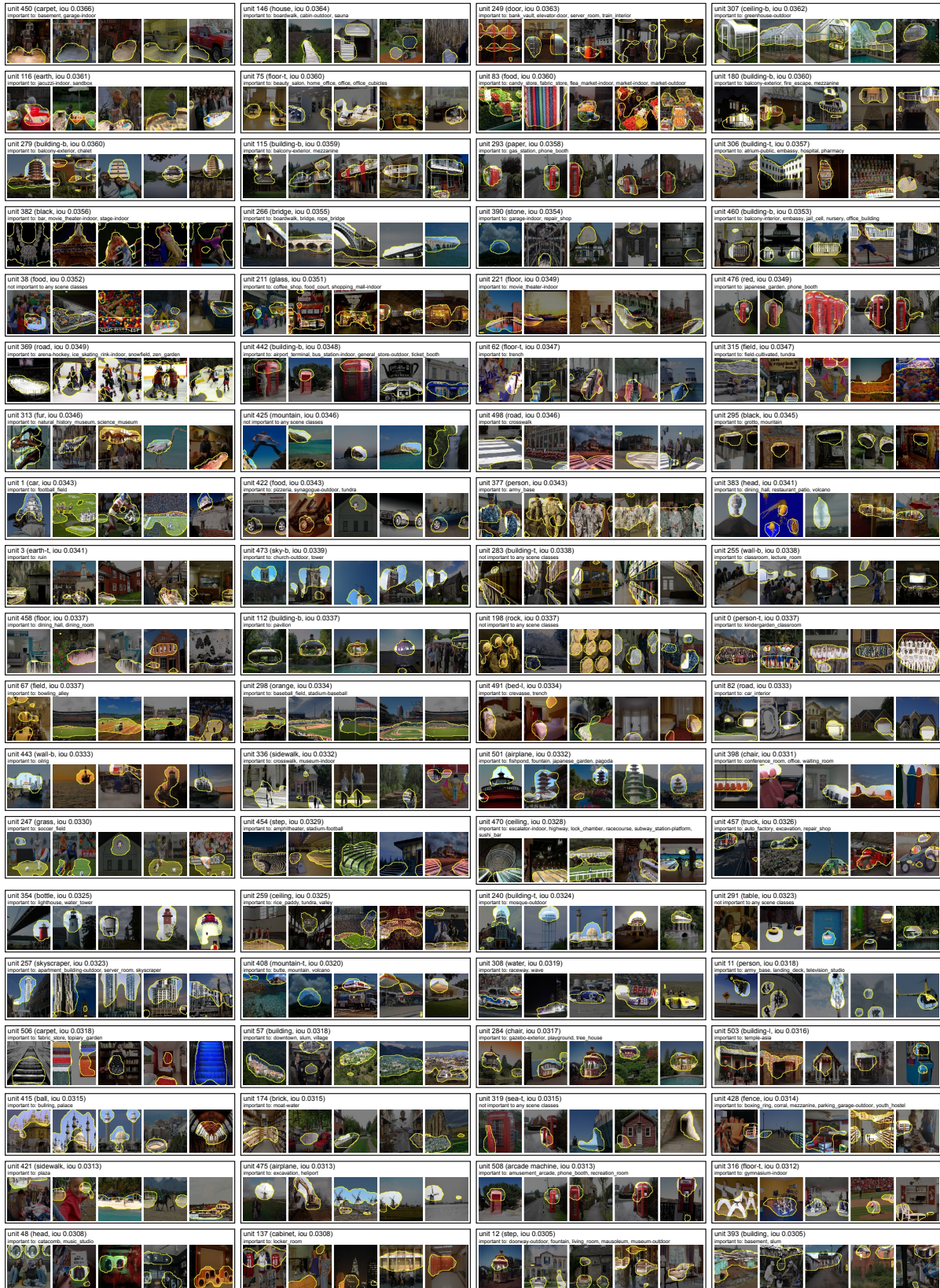


Fig. S10. Comparing all units of conv5_3 of VGG16, 5 of 7.

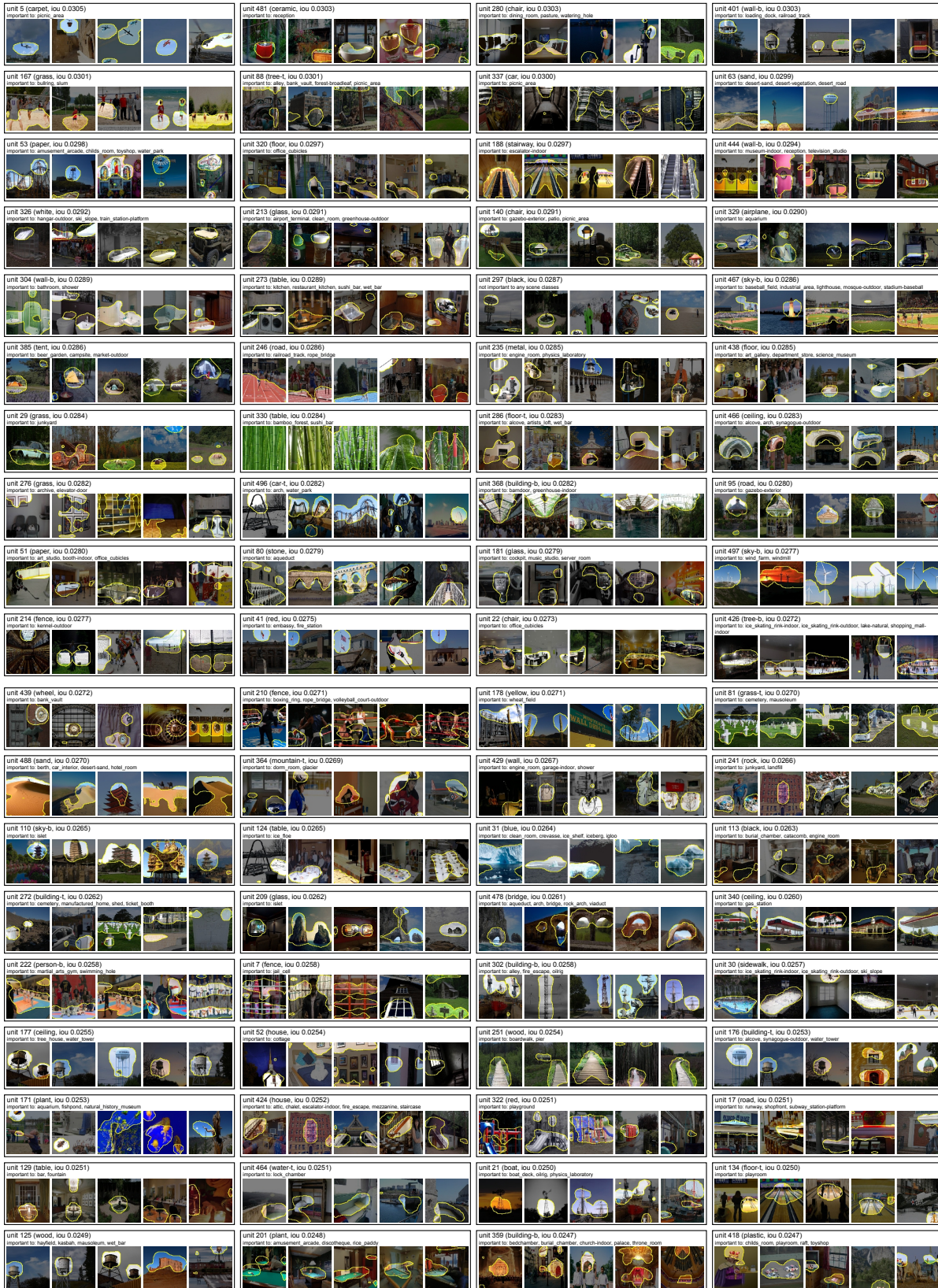


Fig. S11. Comparing all units of conv5_3 of VGG16, 6 of 7.

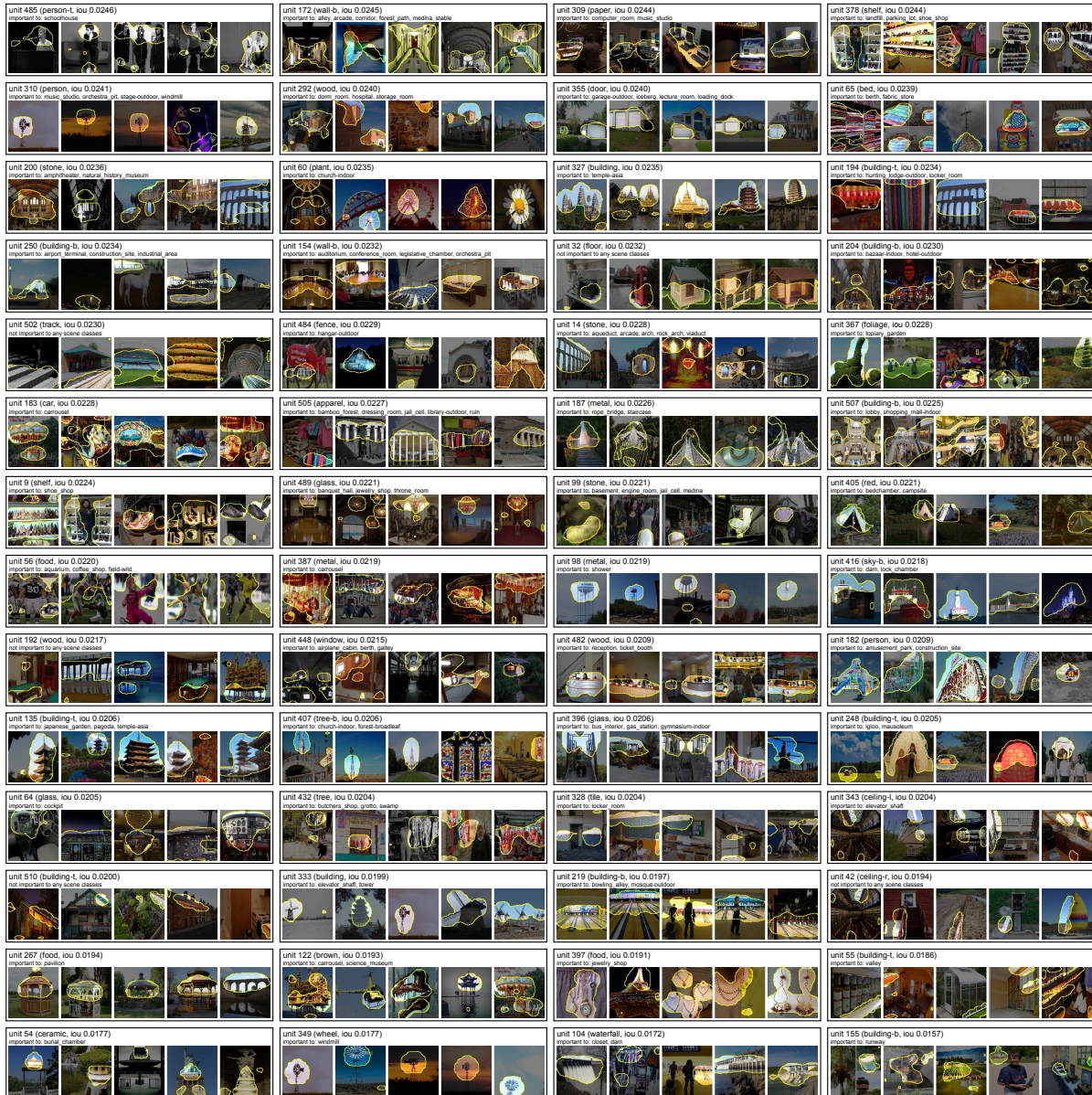


Fig. S12. Comparing all units of conv5_3 of VGG16, 7 of 7.

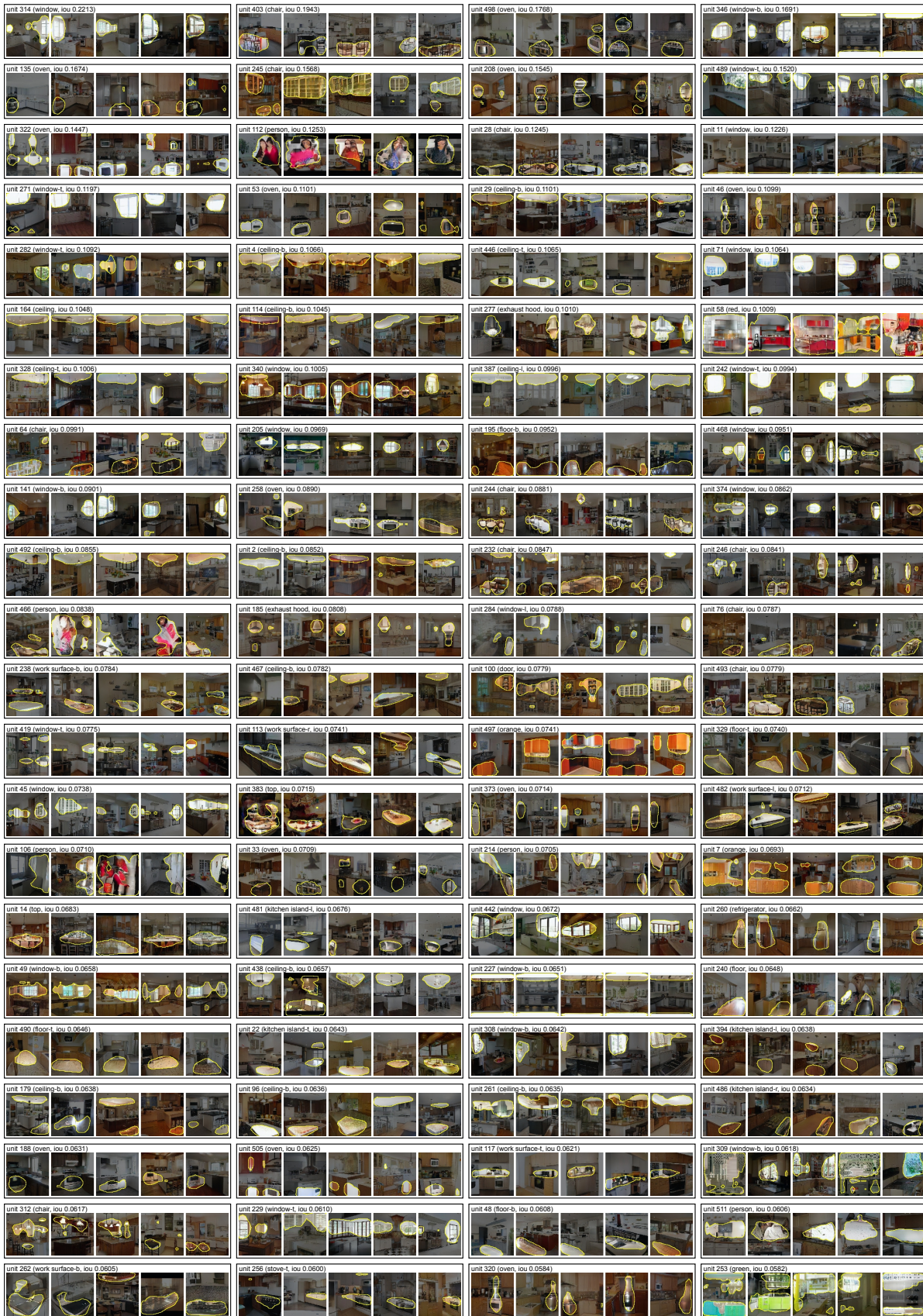


Fig. S13. Comparing all units of $1_{\text{layer}5}$ of Progressive GAN kitchen model, 1 of 6. For each unit, the top five activating images are shown from a sample of 10,000 generated images, highlighting regions where the activation spikes above the top 1% quantile value. Units are sorted in descending order of IoU agreement with a segmented concept.

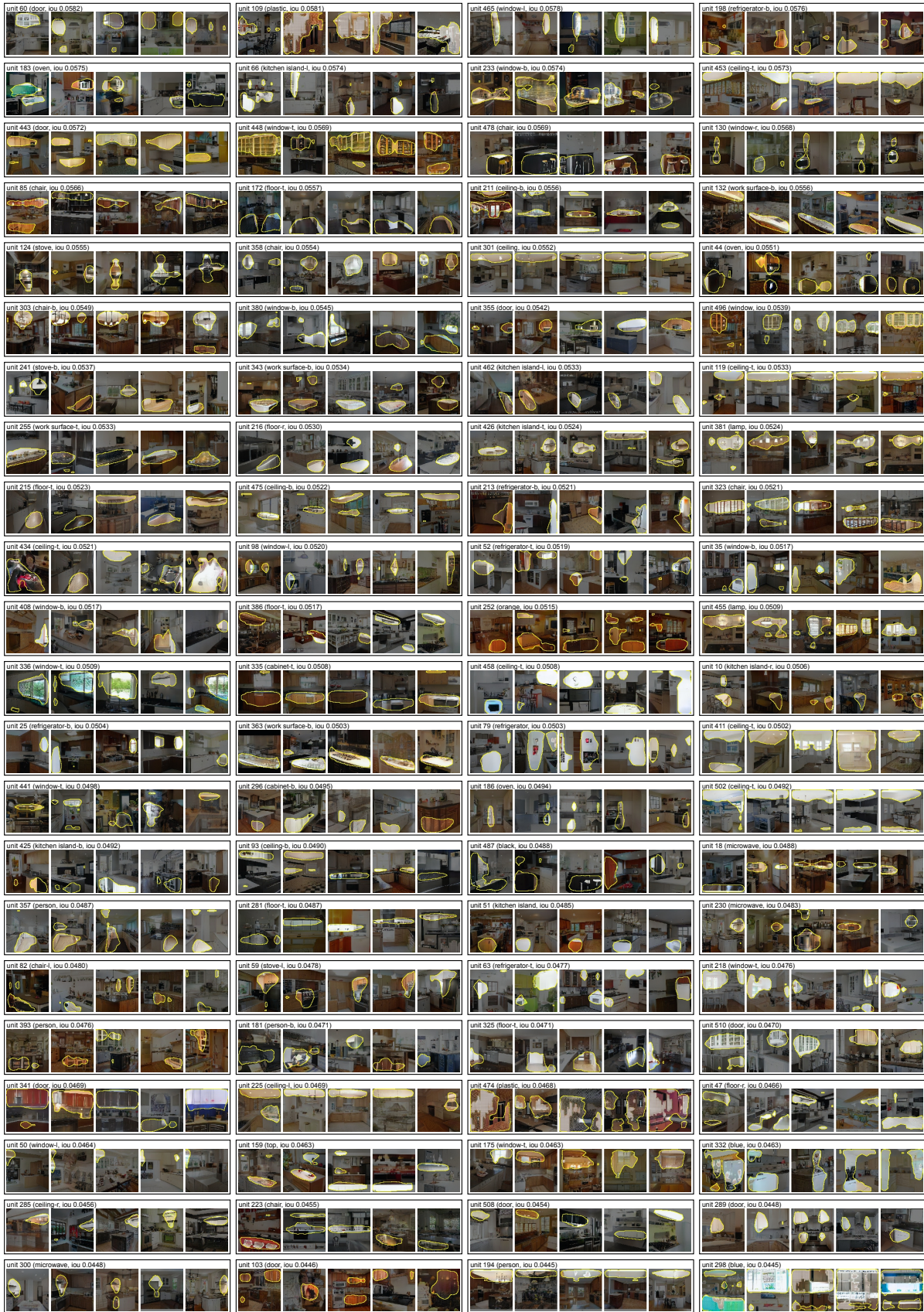
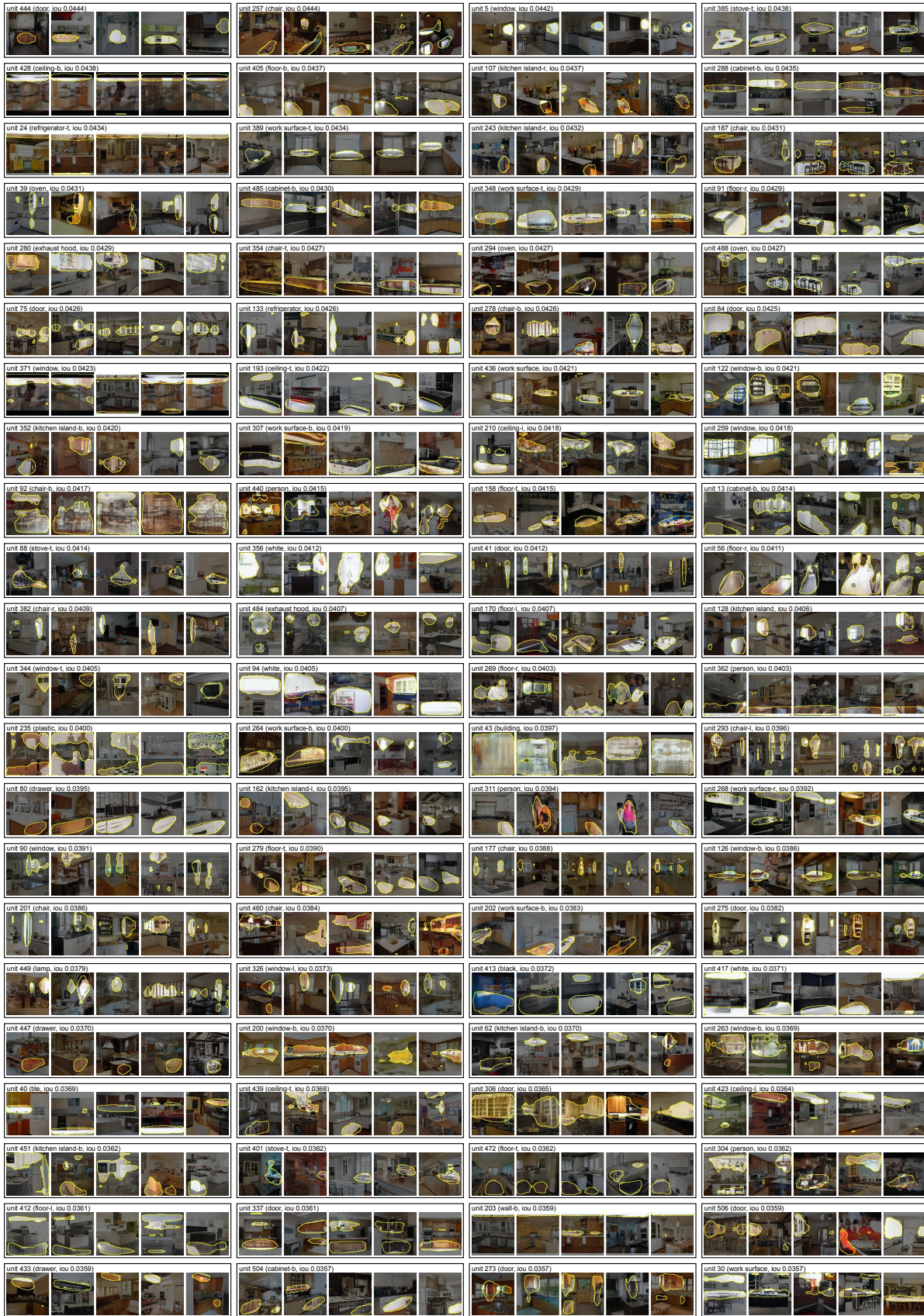


Fig. S14. Comparing all units of 1ayer5 of Progressive GAN kitchen model, 2 of 6.



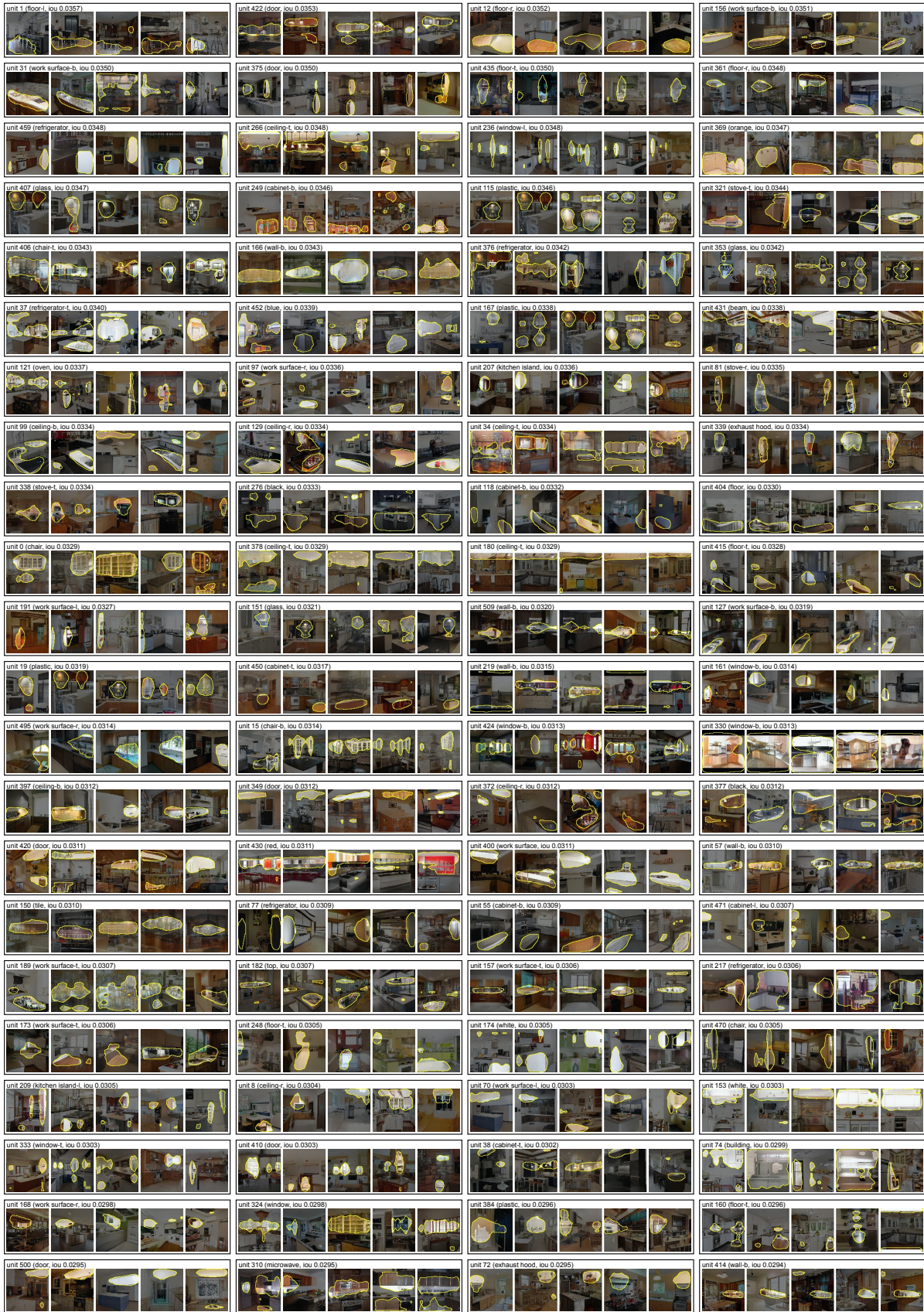


Fig. S16. Comparing all units of layer5 of Progressive GAN kitchen model, 4 of 6.

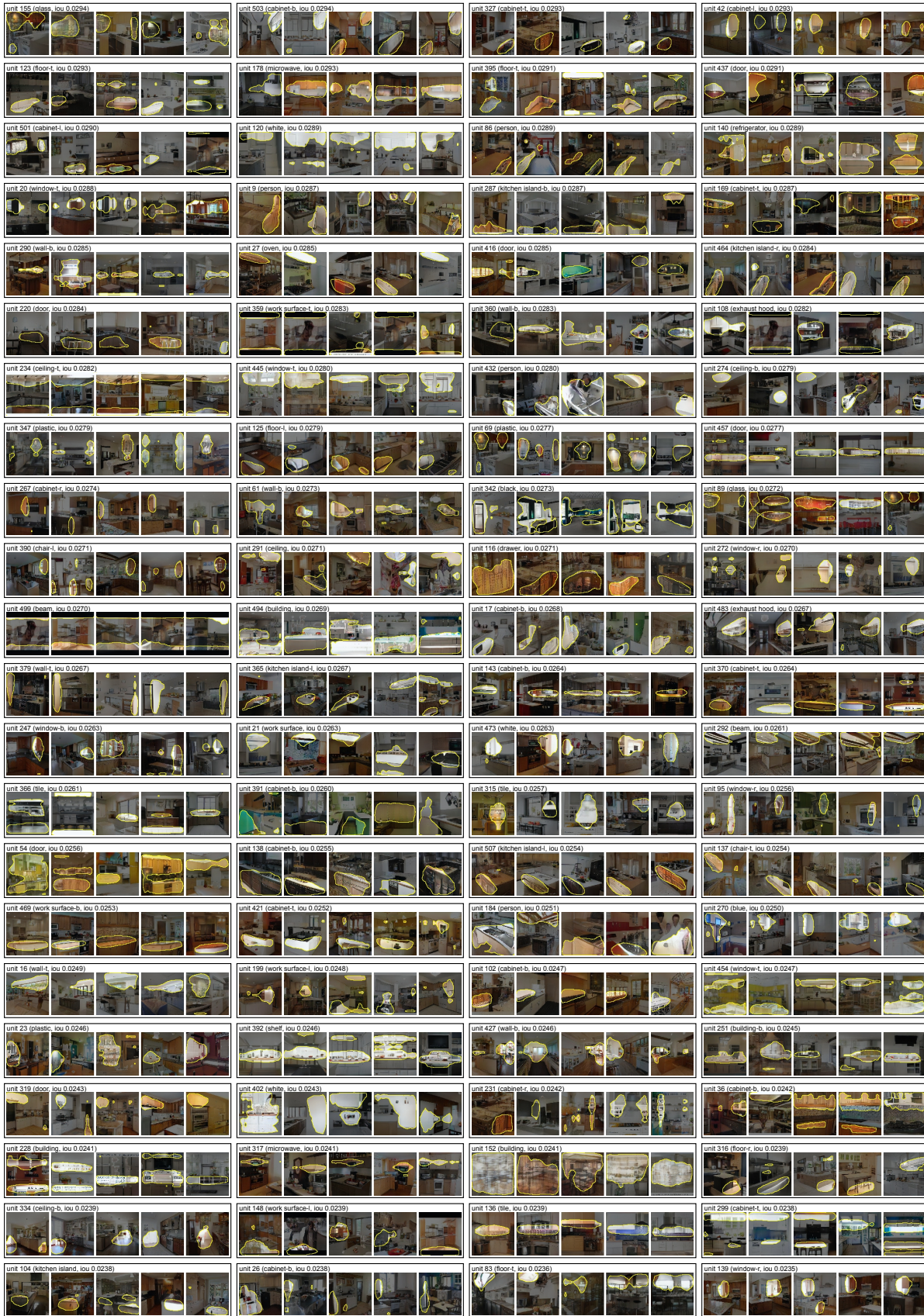


Fig. S17. Comparing all units of 1ayer5 of Progressive GAN kitchen model, 5 of 6.

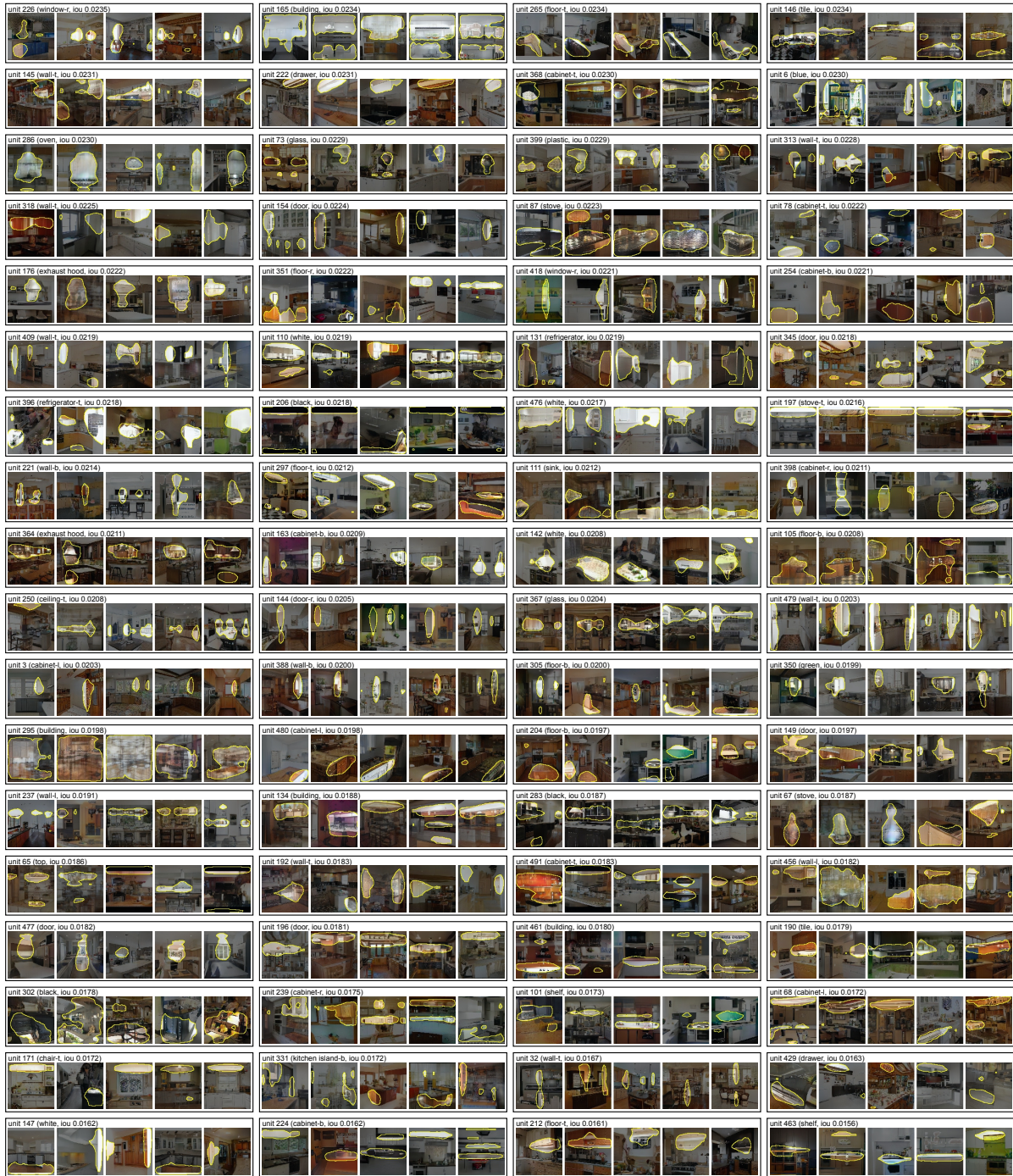


Fig. S18. Comparing all units of layer5 of Progressive GAN kitchen model, 6 of 6.

95 **Movie S1.** Shows the use of an interactive tool to deactivate tree units in a Progressive GAN while leaving
96 the other units unchanged. When tree units are zeroed, trees are erased from the generated scene, revealing
97 details of buildings that were previously occluded by the trees. The ability of the model to render details
98 that are not drawn by default strongly suggests that the GAN has a high-level model of buildings that can
99 be occluded by trees, not just memorized pixel patterns for scenes.

100 **Movie S2.** Shows the use of an interactive tool to activate door units of a Progressive GAN at different
101 locations. Activating door units in different locations produces results that depend on the context. Doors
102 can be added in contexts where a door would be reasonable, in appropriate locations of a building, but they
103 are difficult to add in locations that would not make sense, such as in the sky or in a tree.

104 **Movie S3.** Demonstrates activating and deactivating a variety of GAN units in combination to paint semantic
105 concepts on an image for artistic effect.